Performance of Multiple TCP Connections over Different Routing Protocols in Mobile Ad-hoc Networks

Abhishek Kumar, Ritesh Sood, Huzur Saran & Rajeev Shorey
IBM India Research Laboratory,
Block 1, Indian Institute of Technology,
Hauz Khas, New Delhi 110016, India.
Email: srajeev@in.ibm.com
Phone: 91-11-6861100; Fax: 91-11-6861555

Abstract— **An ad hoc network is a collection of wireless mobile nodes dynamically forming a temporary network without the use of any existing network infrastructure or centralized administration. Various ad-hoc routing protocols have been proposed in literature such as Destination Sequenced Distance Vector (DSDV), ad-hoc On-demand Distance Vector (AODV) and Dynamic Source Routing (DSR). TCP/IP is the most widely used transport protocol for data services such as file transfer (ftp), email and www browsing. Due to these reasons, its use over mobile ad-hoc networks is a certainty. TCP primarily designed for wireline networks, faces performance degradation when applied to the ad-hoc scenario. Earlier work focussed on comparing the performance of different routing protocols with a single TCP connection. In this paper, we study the performance of mutiple TCP connections over various ad-hoc routing protocols. The performance metrics of interest are the TCP throughput and the coefficient of fairness.**

Keywords—**Ad-hoc, Wireless Networks, TCP, Routing, DSR, DSDV, AODV, Fairness coefficient**

I. INTRODUCTION

With the ever increasing demand for connectivity, the need for mobile wireless communication is inevitable. The use of portable laptops and hand held devices is on the rise. Most of the portable communication devices have the support of a fixed base station or access point that corresponds to the last-hop wireless model. This trend can be observed in wide area wireless cellular systems and indoor pico cellular systems such as the Bluetooth technology. However, such a support is not available in settings where access to wired infrastrusture is not possible. Situations like natural disasters, conferences and military settings are noteworthy in this regard. This has led to the development of *mobile ad-hoc networks.*

An ad hoc network is a dynamically changing network of mobile devices that communicate without the support of a fixed infrastructure. There is a direct communication among neighbouring devices but communication between non-neighbouring devices requires a routing algorithm. A lot of work has been done on routing protocols since they are critical to the functioning of ad-hoc networks. Various Routing Protocols have been proposed in literature such as Destination Sequenced Distance Vector (DSDV), Adhoc On-demand Distance Vector (AODV) and Dynamic Source Routing (DSR) [1], [2].

TCP is the most widely used transport protocol for data services such as file transfer (ftp), email and www browsing. Due to these reasons, its use over mobile ad-hoc networks is a certainty. TCP primarily designed for wireline networks, faces performance degradation when applied to the ad-hoc scenario. Prior work in this area has focussed on comparing the performance of a single TCP connection over different routing protocols. The authors in [3] study the performance of TCP over different ad hoc routing protocols. Their study includes the signal stability based adaptive routing (SSA). A detailed study on the performance of TCP over various ad hoc routing protocols can be found in [4].

In an ad hoc network with multiple devices, it is reasonable to expect that there will be multiple TCP connections simultaneously. A realistic study of TCP over ad hoc routing protocols should include not only one TCP connection but several TCP connections over an ad hoc network of N nodes. This paper presents the results of a detailed packet-level simulation studying the performance of multiple TCP connections over various ad hoc routing protocols: DSDV, DSR and AODV. The performance metric of interest is the *fairness coefficient* and the *TCP throughput.*

This paper is organized as follows. Section II gives a brief introduction to some of the well studied ad-hoc routing protocols. In Section III, we describe the simulation model with its parameters. In Section IV, we present our simulation results and analyses. Finally, we conclude by presenting a summary of our results and some suggestions for future work in Section V.

II. OVERVIEW OF AD-HOC ROUTING PROTOCOLS

We now discuss DSR, DSDV and AODV routing protocols in brief. DSDV is a *table-driven* routing protocol. These protocols require each node to maintain one or more tables to store routing information, and they respond to changes in the network topology by propagating updates throughout the network in order to maintain a consistent network view. DSR and AODV protocols belong to the family of *source-initiated on-demand* routing protocols. This type of routing creates routes only when desired by the source node.

A. Dynamic Source Routing

Dynamic Source Routing (DSR) employs "source routing" wherein the source determines the complete sequence of nodes through which a packet has to be routed. Whenever a source has a packet to transmit, it checks its routing table for a route to the destination. In case a route is not found, a route request (RREQ) broadcast is initiated. On receiving this request, each node again broadcasts this request by appending its address to the request packet until this packet reaches the destination. The destination replies to the first request that reaches it. It sends a route reply (RREP) to the source containing the route from the source to the destination. When this packet reaches the source, a connection is established and all subsequent packets contain the complete route in the packet header. No routing information is maintained at the intermediate nodes. When the data link layer at a particular node encounters a transmission failure, it issues an error notification to the source and a new route search is initiated.

B. Destination Sequenced Distance Vector routing

In Destination Sequenced Distance Vector routing (DSDV), each node maintains a routing table wherein the next-hop information for each reachable destination is maintained. Every node in the network periodically broadcasts its routing table with monotonically increasing sequence numbers. An update is done using the Bellman-Ford algorithm. A broken link can be detected if no broadcasts have been received from a node for a while. On detection of a broken link, all routes passing through that hop are assigned an "infinity metric".

C. Ad-hoc On-demand Distance Vector routing

Ad-hoc On-demand Distance Vector routing algorithm borrows its salient features from DSR and DSDV. When a source needs a path to a destination, it broadcasts a route request message enclosing a monotonically increasing "broadcast id" and the last known sequence number to that destination. The route request is broadcast until it reaches a node that has a route to the destination with a destination sequence number higher than that enclosed in the request. A route request propagating through the network establishes the next hop information for the reverse route to the source. A route reply generated by the destination propogates along the reverse route and establishes the forward route information at the intermediate nodes. Each node records only the next hop for a destination and not the entire route as done in source routing protocols. Routing table information in AODV is restricted to the active nodes. A neighbour is considered active if it originates or relays at least one packet for the destination within the most recent "active timeout" period. Faliure of a link can be detected via *hello* messages or link layer detection. When a link goes down, the upstream nodes are notified of the faliure and that destination is marked as unreachable in the routing tables of these nodes.

III. SIMULATION MODEL

In this paper we have used simulations to study the performance of multiple TCP connections over ad-hoc routing protocols. We have carried out the simulations in Network Simulator (NS-2) from Lawrence Berkeley National Laboretory (LBNL) [8] with extensions for wireless links from the Monarch project at Carnegie Mellon University.

At the physical layer, the radio propagation model chosen emulated a propagation delay, omni-directional antennas, and a shared media network interface, whose physical characteristics parameters were initialised to emulate the 914 MHz Lucent WaveLAN DS-SS radio interface. The IEEE 802.11 Medium Access Protocol is used at the Link Layer.

For the performance analysis of the routing protocols, we have simulated a square topology of 500m x 500m. The radio interface emulated has a nominal transmission range of 250m and a bandwidth of 2Mbps.

We have employed multiple FTP applications running over TCP connections in the simulations. The version of TCP used is TCP Tahoe [6], [5]. The nodes in the simulations move according to the "random waypoint" model. The speed of the mobile nodes is a uniformly distributed random variable between 0 to 5 m/s. At the start of a simulation, each node randomly selects its destination and moves towards it. On reaching its destination, it again chooses a random destination and repeats the procedure till the simulation ends at 100 nanoseconds.

A predetermined number of nodes start FTP applications (over TCP) at a randomly selected instant between 0 to 10 seconds. The throughput of each connection from the 15th second to the end of simulation at 100 seconds is measured. Simulations have been run for 1, 4, 8, 14 and 20 TCP connections. For a given number of TCP connections, 50 simulation instances with 50 randomly generated mobility scenarios have been run and the average of the results taken. The above process is repeated with 15, 20 and 30 nodes in the topology and using each of the three routing protocols, i.e., DSR, DSDV and AODV. In our simulations, the TCP packet size is 512 bytes and TCP ACK size is 40 bytes.

A. *Performance Metrics*

The performance metrics that we have studied are (i) the TCP throughput and (ii) the coefficient of fairness. *Throughput* is an indication of how much data the user can receive per second. In our simulation results, the throughput is measured as the total number of non-duplicate TCP packets received between the 15th to the 100th second of the simulation. The results of the first 15 seconds are not considered in order to avoid initialization bias (during the startup of TCP connections). We define the *cumulative throughput* to be the sum of throughput of individual TCP connections over all such connections.

For the protocol to be fair, each TCP connection should get almost equal throughput. Denoting x_i to be the throughput of connection i, $i = 1, 2, \dots, n$, we define the fairness index (or the coefficient of fairness) as follows [5] (page 401),

$$f(x_1, x_2, \dots, x_n) = \frac{(\sum_{i=1}^{n})^2}{n \sum_{i=1}^{n} x_i^2} \qquad (1)$$

The fairness index always results in a number between 0 and 1. If all throughputs are the same, the fairness index is 1.

To compute the fairness index, we take the average over 50 simulation runs. Further, *weighted fairness* has been defined as the coefficient of fairness calculated after multiplying the throughput of individual connections by their respective mean hop-length during the period for which throughput is considered.

IV. SIMULATION RESULTS AND PERFORMANCE EVALUATION

A. *Thoughput*

In Figure 1, Figure 2 and Figure 3, we show the variation of TCP Tahoe cumulative throughput versus the number of TCP connections for the three routing protocols, i.e., DSR, DSDV and AODV respectively. As can be seen from these figures, for the three ad-hoc routing protocols, the throughput initially increases with the number of TCP connections till it reaches a peak and then falls off with a further increase in the number of TCP connections. This behaviour can be explained in terms of a tradeoff between bandwidth usage and routing overheads. Too few TCP connections do not use the available bandwidth completely while a larger than optimum number of TCP connections results in an increased routing overhead and hence a less than optimum throughput.

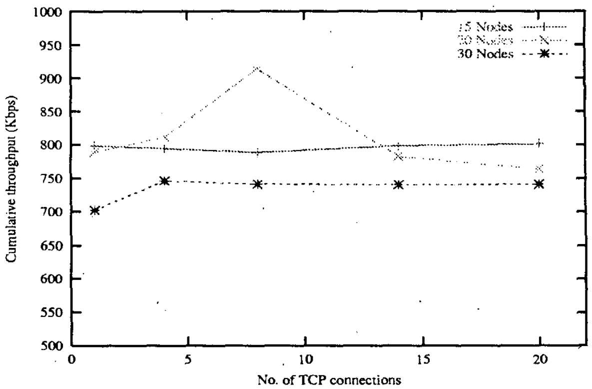

Fig. 1. TCP Throughput vs No. of Connections in DSR

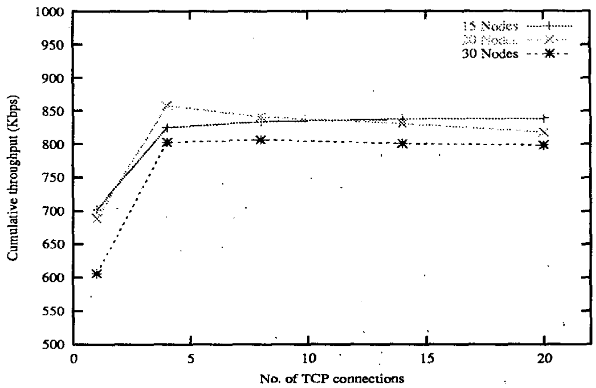

Fig. 2. TCP Throughput vs No. of Connections in DSDV

Further, when the number of TCP connections are large, the cumulative throughput decreases with an increase in the number of nodes in the ad-hoc network. With an increase in the number of nodes, the routing overheads also increase, especially for table driven protocols such as DSDV. Note that the possibility of choosing a non-optimal route increases, thus increasing the average hop length which results in a degradation of performance.

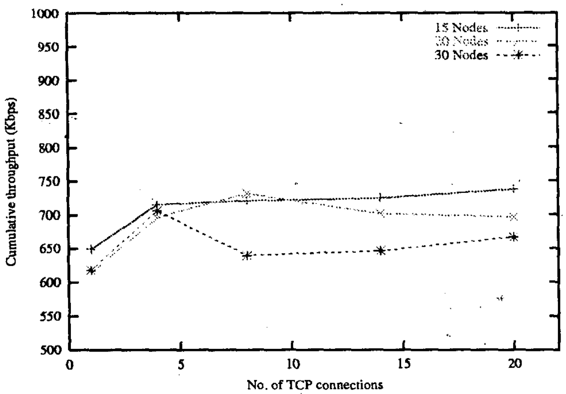

Fig. 3. TCP Throughput vs No. of Connections in AODV

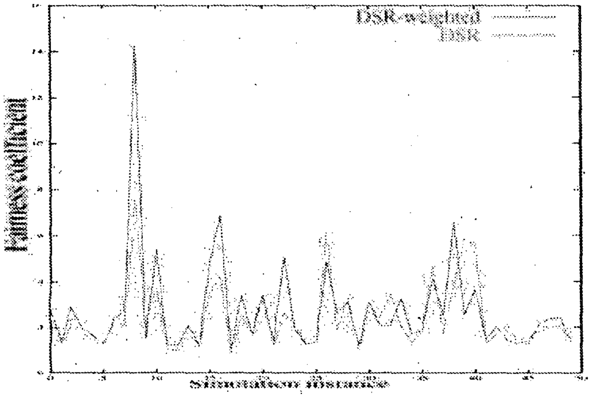

Fig. 4. Fairness Coefficient with Varying Simulation Instances with DSR

B. Fairness

In Figure 4 and Figure 5, we plot the coefficient of fairness and weighted fairness for the 50 simulation runs for DSR and DSDV ad-hoc routing protocols respectively. Each simulation run corresponds to a randomly chosen mobility pattern. Further, it is seen that the fairness of wireless ad-hoc networks is significanty less than that of wireline LANs. Upon simulating a LAN in which 4 TCP connections share the same channel, the fairness coefficient is greater than 20, while the average fairness coefficient in an ad-hoc networks is found to be below 5; it rarely exceeds 10. The plot for weighted fairness envelopes the plot for coefficient of fairness thus indicating that fairness experienced by individual conections reduces with an increase in hop-length for that particular connection.

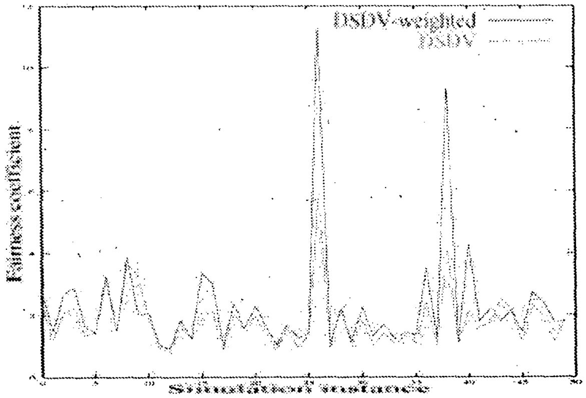

Fig. 5. Fairness Coefficient with Varying Simulation Instances with DSDV

V. CONCLUSION

This paper presents the results of a detailed packet-level simulation studying the performance of multiple TCP connections over various ad hoc routing protocols: DSDV, DSR and AODV. The performance metrics of interest are the *fairness coefficient* and the *TCP throughput*. A high value of fairness coefficient suggested that the throughput across multiple TCP connections is almost equal, a low value implying that some TCP connections have a high throughput at the expense of other connections.

REFERENCES

[1] E. M. Royer and Chai-Keong Toh, "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks", *IEEE Personal Communications*, April 1999.

[2] J. Broch, D. A. Maltz, D. B. Johnson, Y. Hu and J. Jetcheva, "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols", *ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'98)*, October 1998, Dallas, Texas, USA.

[3] A. Ahuja, S. Agarwal, J. P. Singh and R. Shorey, "Performance of TCP over Different Routing Protocols in Mobile Ad Hoc Networks", *IEEE Vehicular Technology Conference (VTC)*, May, 2000, Tokyo, Japan.

[4] G. Holland and N. H. Vaidya, "Analysis of TCP Performance over Mobile Ad Hoc Networks", *ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99)*, Seattle, WA, USA, August 1999.

[5] L. L. Peterson and B. S. Davie , *Computer Networks: A Systems Approach*, Morgan Kaufmann Publishers, Inc, 1996.

[6] W.R. Stevens, *TCP/IP Illustrated, Volume 1: The Protocols*, Addison Wesley, 1994.

[7] K. Fall and S. Floyd, "Simulation-based Comparisons of Tahoe, Reno, and Sack TCP," *ftp://ftp.ee.lbl.gov*, Mar. 1996.

[8] S. McCanne and S. Floyd, "NS-Network Simulator," 1995, http://www-nrg.ee.lbl.gov/ns